

Generalized Quantitative Analysis of Metric Transition Systems

Uli Fahrenberg Axel Legay

IRISA/INRIA Rennes, France

MEALS Momentum Gathering 2013

Elevator Statement

When formal models include **quantities**,
the standard Boolean relations
such as simulation, language inclusion, bisimulation, etc.
have **little use**.

They need to be replaced by **distances**.
There is, however, a lot of disagreement
how precisely to do this,
so a **unifying metric theory**
of **quantitative analysis**
is called for.

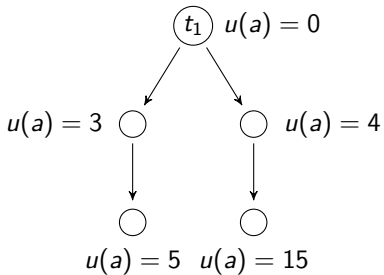
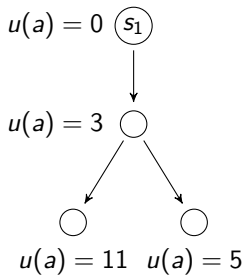
- 1 Metric Transition Systems
- 2 Distances
- 3 General framework for system distances
- 4 Conclusion

Metric Transition Systems

Quantitative model du jour:

- **metric transition system**: $(S, T, [\cdot])$, with $[\cdot] : S \rightarrow \mathcal{U}[\Sigma]$
- Σ : atomic propositions; $\mathcal{U}[\Sigma]$: set of **valuations** $u : \Sigma \rightarrow X$
- (X, d) : (extended) hemimetric space
- (hemimetric: **asymmetric pseudometric**)
- essentially the setting from [Alfaro, Faella, Stoelinga: Linear and branching system metrics, IEEE Trans. Softw. Eng. 35(2):258–273, 2009]

Example



Distances

Propositional distance:

$$pd(u, v) = \sup_{a \in \Sigma} d(u(a), v(a))$$

State distance:

$$pd(s, t) = pd([s], [t])$$

- **syntactic** distance between states
- want: **semantic** distance between states' **behaviors**

Measuring distances between behaviors

- behavior = **trace** (finite or infinite)
- point-wise** trace distance:

$$td(\sigma, \tau) = \begin{cases} \infty & \text{if } \text{len}(\sigma) \neq \text{len}(\tau), \\ \sup_i pd(\sigma_i, \tau_i) & \text{otherwise.} \end{cases}$$

- discounted accumulating** trace distance ($\lambda \in]0, 1[$):

$$td(\sigma, \tau) = \begin{cases} \infty & \text{if } \text{len}(\sigma) \neq \text{len}(\tau), \\ \sum_i \lambda^i pd(\sigma_i, \tau_i) & \text{otherwise.} \end{cases}$$

- limit-average** trace distance:

$$td(\sigma, \tau) = \begin{cases} \infty & \text{if } \text{len}(\sigma) \neq \text{len}(\tau), \\ \liminf_j \frac{1}{j+1} \sum_{i=0}^j pd(\sigma_i, \tau_i) & \text{otherwise.} \end{cases}$$

- and a bunch of others, all with their own reasonable motivation

From behavioral distance to semantic state distance

[AFS09] consider only (discounted) point-wise distance:

- trace distance (recall): $td(\sigma, \tau) = \sup_i pd(\sigma_i, \tau_i)$
- **linear** distance:

$$ld(s, t) = \sup_{\sigma \in \text{Tr}(s)} \inf_{\tau \in \text{Tr}(t)} td(\sigma, \tau)$$

- generalizes **trace inclusion**; has symmetric cousin
- **branching** distance: least fixed point to

$$sd(s, t) = \sup_{s \rightarrow s'} \inf_{t \rightarrow t'} \max\{sd(s, t), sd(s', t')\}$$

- generalizes **simulation**; has symmetric cousin

How to generalize this to all the other useful distances?

General framework for system distances

Given: trace distance td . **Want:** linear & branching distances ld , sd

- for a set M , let $\mathbb{L}M = M \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$
 - complete lattice; $\alpha \sqsubseteq \beta$ iff $\forall x. \alpha(x) \leq \beta(x)$
 - addition $\alpha \oplus \beta = \lambda x. \alpha(x) + \beta(x)$ (“Girard quantale”)

Definition

A *recursive specification* of a trace distance td consists of

- a set M and a lattice homomorphism $\text{eval} : \mathbb{L}M \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$,
- a hemimetric $td^{\mathbb{L}} : \mathcal{U}[\Sigma]^{\infty} \times \mathcal{U}[\Sigma]^{\infty} \rightarrow \mathbb{L}M$ s.t. $td = \text{eval} \circ td^{\mathbb{L}}$,
- and a **distance iterator** $F : \mathcal{U}[\Sigma] \times \mathcal{U}[\Sigma] \times \mathbb{L}M \rightarrow \mathbb{L}M$.

F must be monotone in the third coordinate and satisfy

$$td^{\mathbb{L}}(u.\sigma, v.\tau) = F(u, v, td^{\mathbb{L}}(\sigma, \tau))$$

Examples of recursive specifications

- point-wise: $M = \{*\}$

$$td(u.\sigma, v.\tau) = \max(pd(u, v), td(\sigma, \tau))$$

- discounted accumulating: $M = \{*\}$

$$td(u.\sigma, v.\tau) = pd(u, v) + \lambda td(\sigma, \tau)$$

- limit-average: $M = \mathbb{N}$

$$td^{\mathbb{L}}(u.\sigma, v.\tau)(j) = \frac{1}{j+1} pd(u, v) + \frac{j}{j+1} td(\sigma, \tau)$$

$$td(\sigma, \tau) = \liminf_j td^{\mathbb{L}}(\sigma, \tau)(j)$$

Examples of recursive specifications

- point-wise: $M = \{*\}$

$$td(u.\sigma, v.\tau) = \max(pd(u, v), td(\sigma, \tau))$$

- discounted accumulating: $M = \{*\}$

$$td(u.\sigma, v.\tau) = pd(u, v) + \lambda td(\sigma, \tau)$$

- limit-average: $M = \mathbb{N}$

$$td^{\mathbb{L}}(u.\sigma, v.\tau)(j) = \frac{1}{j+1} pd(u, v) + \frac{j}{j+1} td(\sigma, \tau)$$

$$td(\sigma, \tau) = \liminf_j td^{\mathbb{L}}(\sigma, \tau)(j)$$

All commonly used trace distances have recursive specifications.

From recursive specification to linear & branching distance

Given: trace distance td with recursive specification $td = \text{eval} \circ td^{\mathbb{L}}$,
 $td^{\mathbb{L}}(u.\sigma, v.\tau) = F(u, v, td^{\mathbb{L}}(\sigma, \tau))$

Definition

The **linear** distance from s to t is

$$ld(s, t) = \sup_{\sigma \in \text{Tr}(s)} \inf_{\tau \in \text{Tr}(t)} td(\sigma, \tau)$$

The **branching** distance from s to t is $sd = \text{eval} \circ sd^{\mathbb{L}}$, with $sd^{\mathbb{L}}$ the least fixed point to

$$sd^{\mathbb{L}}(s, t) = \sup_{s \rightarrow s'} \inf_{t \rightarrow t'} F([s], [t], sd^{\mathbb{L}}(s', t'))$$

Conclusion

- From a **recursive specification** of a trace distance, we get definitions of corresponding **linear** and **branching** distances
- These are generalizations of **trace inclusion** and **simulation**
- Theorem: **always $ld(s, t) \leq sd(s, t)$**
- This generalizes a number of approaches in the litterature
- Similarly one can get: **trace equivalence** distance, **bisimulation** distance, **nested simulation** distance, **ready trace** distance, etc.
- A quantitative linear-time–branching-time spectrum!

- Next step: Transfer this to **probabilistic** automata and relate to prior work in this area